Visibility Cues for Communication Aware Guidance in Cluttered Environments

H. Claus Christmann and Eric N. Johnson

Abstract This paper presents the usage of visibility based guidance cues in order to find waypoints useful for maintaining communication in a multi UAV (Uninhabited Aerial Vehicle), single operator system. Based upon the overlay of visibility graphs (for radio communication) and Voronoi diagrams (for maximum clearance motion paths), the paper presents simulations of three staged methods, allowing the computation of waypoints suitable for establishing a potential multi-hop connection between an operator and a primary UAV in an urban or otherwise cluttered environment. The methods present generic solutions for 2D planes, ensuring applicability for indoor, outdoor, and other structured environments through a potential interconnection of several non-coplanar 2D planes. The presented methods increase in computational complexity as they are capable of handling more complex scenarios. However, the presented methods are overall still deemed computationally acceptable and present themselves as good candidates for onboard implementation on vehicles with limited computational power.

1 Introduction and Motivation

Tactical Uninhabited Aerial Systems (UAS) often utilize a single Uninhabited Aerial Vehicle (UAV) tele-operated by a single control station operator. Though higher level control, i.e. the use of preprogrammed waypoints or whole trajectories, is sometimes possible, the remote operators often pilot the UAV directly via a first-person video feed, providing them with immediate sensor data and allowing them

H. Claus Christmann
Georgia Institute of Technology, School of Aerospace Engineering, Atlanta, GA 30332-0150
e-mail: hcc@gatech.edu

Eric N. Johnson
Georgia Institute of Technology, School of Aerospace Engineering, Atlanta, GA 30332-0150
e-mail: eric.johnson@ae.gatech.edu

to perform tasks such as obstacle detection and classification, collision avoidance, and path planning.[1] These first-person video streams in combination with stability augmentation systems for remote piloting allow for increased ease of operation, high situational awareness of the operator, and direct availability of the primary sensor data, the video feed. All that is achievable with relatively modest training requirements - both for the control station operation part as well as the actual remote-piloting part.

However, this single-operator-single-vehicle setup limits the operational range of such a UAS to essentially the range of the utilized communication link. Furthermore, given, for example, the height of urban high rise buildings, positioning UAVs “above and behind” (as required by Line-Of-Sight (LOS) constraints) might not always allow suitable sensor access to the back side of Radio Frequency (RF) obstructing objects. Using a relay can mitigate those limits. More complex UAS, e.g. current High-Altitde-Long-Endurance (HALE) or Medium-Altitude-Long-Endurance (MALE) systems, can utilize indirect communication via communication relay nodes to overcome this LOS limit, most often at a cost of link delay and the addition of an operator dedicated to payload related activities.

For tactical scale UAS the use of satellites as relays is prohibitive, not only due to the introduction of high latency, but foremost for infeasibility of the implementation of related required avionics. Relying on potentially available HALE or MALE systems to act as relays is also challenging. Not only would the local tactical UAS operator have to coordinate with a different UAS to negotiate operational areas and coverage, but also would the link to the relay HALE or MALE UAV have to be robust to shadowing and/or multi-path effects in cluttered urban environments.

Instead of external pseudolites, other local UAVs from within the same tactical UAS could be used as communication relay nodes, effectively establishing a local multi-hop network within the UAS.

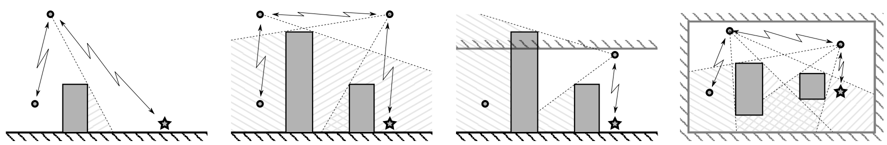
However, if operated under the same principle of remote-piloting, introducing additional UAVs as relays comes at the cost of drastically increased workload. Each additional relay UAV would require a similar amount of work as the primary UAV, mainly work related to collision avoidance and path planning. For those secondary UAVs, path planning is furthermore complicated by the dual task of getting from one location to another as well as maintaining LOS to the primary UAV as well as to the GCS or other intermediaries, respectively.

This work proposes visibility based cues that could allow secondary UAVs to conduct these relay tasks without major operator intervention, combining operational advantages of smaller scale tactical UAS with the benefits of swarm-enabling, higher-level automation in the background.

1.1 Limiting the Operational Zone of the Secondary UAVs

Starting from the HALE and MALE analogy, an initial replication of such a setup seems suitable. The system would deploy a single supportive UAV as a relay, this

UAV would position itself "high and behind" any potential obstacle, Fig. 1(a), and as such establish a dual-hop link. An extension to this would be the usage of two supportive UAVs, positioned high enough directly above the control station and the primary UAV, respectively, to establish a three-hop link, Fig. 1(b). These setups could conceptually be called *vertical*, as the task involves the placement of supportive UAVs in an essentially vertical plane determined by the position of the control station, the primary UAV, and the "up"-direction.



(a) Placing a single supportive UAV high and behind obstacles in cases of few obstacles.

(b) Supportive UAVs high and directly over other nodes helps in denser environments.

(c) The vertical approach could fail if going high is not an option for some reason.

(d) Utilizing a horizontal setup could enable otherwise infeasible solutions.

Fig. 1 Open environments pose no special problems as LOS is essentially guaranteed. In the presence of RF obstacles, using a vertical 2D plane to create multi-hop links between the control station and the primary UAV provides for conceptually identical solutions. If the scenario does not allow for such a positioning, using a horizontal 2D plane can expand the solution space and enable previously not possible setups.

However, "going up" might not always be an option. In certain scenarios the airspace could be closed above a given altitude or there is no LOS between a primary UAV and the space above, potentially due to the Area of Interest (AoI) being in a tunnel, under a large bridge, indoors, or under ground, Fig. 1(c). To include such scenarios, the solution would have to include the *horizontal* component of the environment, Fig. 1(d). However, as the problem is still a 2D problem, the proposed generic processes can also solve this setup.

In both scenarios, the *vertical* as well as the *horizontal* one, a large part of the problem can be captured in a, respectively, vertical or horizontal 2D plane. As a lot of human created environments tend to be "2.5D" - two dimensional complexes or mazes extruded in the "up"-direction and then stacked on top of each other - the operational zone of the supportive UAVs has been chosen to be limited to spaces representable[1] by a 2D plane. This provides a generic solution for the *vertical* and the *horizontal* setup and allows an extension into structured 3D environments by dissecting the environment into a set of mutually intersecting 2D planes, Fig. 2.

[1] In order to be allowable, the utilized projection has to maintain the visibility property of the mapped points, i.e. a simple top-down view is only permissible in the absence of larger hills, etc.

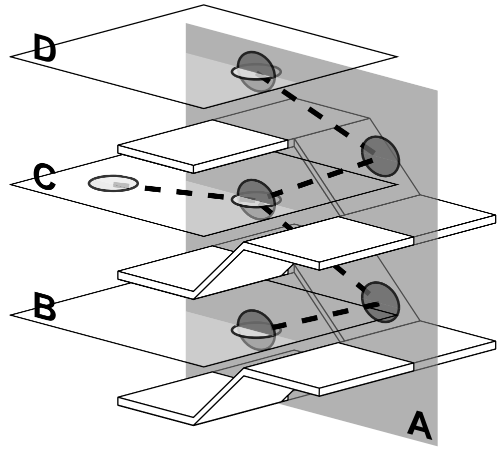


Fig. 2 In complex structured 3D environments, the generality of the solution for the 2D plane case allows for an intersecting of planes to capture the environment. In the displayed example (stylizing a staircase), one horizontal plane captures each floor (B,C,D), a single vertical plane in the staircase joins them together (A). Requiring a UAV to be located at each plane intersection allows inter-plane communication based on the processes presented for the generic 2D planar case.

1.2 Urban First-Responder Scenario

To further motivate the application, an urban first-responder scenario is proposed.[2] In this scenario, urban first responders are assumed to have access to a tactical UAS to support their mission. The first-responders would be accompanied by a UAS operator that manages the UAS, gathers mission relevant information through it, and distributes the gathered data to the affected members of the team.

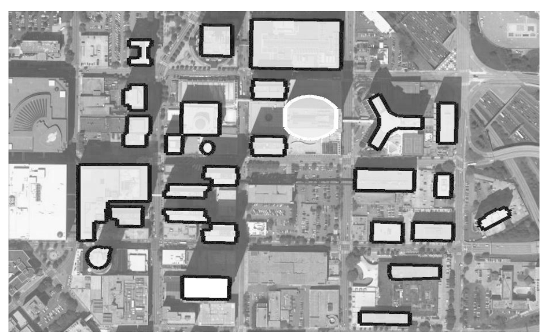


Fig. 3 An urban first-responder scenario: the UAS operator is tasked with an reconnaissance type mission on the target building (white border), i.e. a fire on the 13th floor. Relevant obstacles are also highlighted (black borders). (Aerial Image: Google)

In the scenario a designated primary UAV would be under complete operator control at all times, providing the before mentioned benefits. The (additional) secondary UAVs (acting as relay stations) would be fully autonomous. The operational zone of the secondary UAVs would be a horizontal 2D-plane at a predefined altitude, assuming that the UAVs can't go high enough to clear the buildings. This also supports the system's predictability for the operator, reducing operator workload by eliminating

[2] For more details on the scenario and the motivation see [2].

questions like “What is it doing?” and “Why is it doing that?.” In Fig. 3 a possible control station screen is presented. The mission target is highlighted and has a white border, physical obstacles protruding the operational zone are also highlighted and have a black border. The task at hand would be to gather visual information from all sides of the target area of interest, in this case the white outlined target building.

The scenario assumes *a priori* availability of a map as this is required for the proposed processes. Sec. 2.1 elaborates how this fits into the overall operational scenario.

2 Motion Map

Motion planning is intricately connected to the map that is used and there is extensive research on the topics of map generation as well as motion planning with urban environments being a focus area not only since the DARPA Urban Challenge. In his dissertation [3], Wooden comments on König and Likhachev [4] by stating that *“optimal planning is outweighed by the need for a “good” plan now.”*

Looking for a “good plan now”, ease of computation is a major driver in choosing methods and algorithms and, as mentioned in Sec. 1.1, limiting the operational area from a full 3D environment to a (vertical or horizontal) 2D plane is a big benefactor. Also, as the operational envelope of the (tele-operated) primary UAV is not at all affected through this constraint, the applicability of the proposed methods to urban, indoor, or otherwise structured environments is maintained, particularly if modeling techniques as outlined in Fig. 2 are utilized.

Limiting other driving requirements to the most fundamental one, collision free motion, generalized Voronoi diagrams present themselves as an immediate candidate to cover the motion aspect in 2D scenarios.

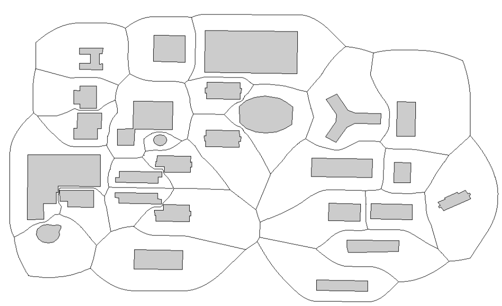


Fig. 4 Voronoi maximum clearance paths through the environment. The graph has been stripped of leaf nodes, resulting in a purely cyclic graph that segments the environment in one connected cell per obstacle. No dead ends allow for easier usage by non-hover capable aircraft, such as fixed wing Micro-UAVs.

Given a *free/occupied* classification of the environment, e.g. Fig. 3, a Voronoi diagram providing maximum clearance paths through this environment can be easily computed. Held’s VRONI [5] provides a computational efficient algorithm to generate Voronoi paths through polygonal environments and the authors believe this algorithm to be suitable for UAV onboard implementation and use.

In order to generate a basic map of permitted paths for the supportive UAVs, the Voronoi graph is stripped of leaf nodes, leaving a dead end free completely cyclic graph, Fig. 4, of presumably collision free paths. The graph also segments the environment into regions associated with each motion obstacle. The borders of the cell encloseing the AoI will later be called (Voronoi) perimeter.

2.1 A priori Data

The proposed processes conceptually fit into the *guidance* category, leaving *navigation* (and the related mapping), and *controls* to other systems. As a result from that stems the requirement for *a priori* availability of a map of the environment, i.e. a *free/occupied* classification. For first-responders in metro areas, these maps are assumed to be made available or, if not, created by the operator during ingress.[3] Furthermore, the processes assume the availability of onboard collision avoidance mechanisms which could report a mismatch between the *a priori* given map and the sensed environment and trigger a conflict resolution procedure to synchronize the map with the sensor information and react appropriately.

As such, the underlying assumption is that the navigation and related mapping problem has been solved, either conventionally through a GPS corrected INS solution or, for example, through a SLAM based approach as in [6].

3 Guidance Cues

Assuming that the supportive UAVs are fully autonomous, the scenario poses a guidance problem: where to send the supportive UAVs to and how to get them there. The guidance task is to propose waypoints which are beneficial for the establishment of a Mobile Ad-hoc NETwork (MANET) in a cluttered environment.

Several researchers have presented results on how to form and maintain MANETs with UAVs (e.g. [7, 8, 9]), though one of the basic assumptions in the presented research is a free space assumption under which the establishment of a link between two nodes depends mainly on the distance of the nodes.[4]

Starting from an identical initial task - getting the primary UAV on the far side of a building - three methods to obtain guidance cues for where to position secondary UAVs are proposed.

---

[3] Given readily available geo-referenced aerial images, the operator could use a familiarity with the area to quickly "click together" a polygonal 2D *free/occupied* classification of the presumed operational area. Bounds on how close to approach these obstacles could be made conservative.

[4] In [10] the authors have proposed the usage of UAS reference scenarios to evaluate the performance of MANET protocols in a free-space situation.

Without any relay nodes and the assumption that the control station operator is be stationary during an active use of the primary UAV, the operational range of the primary UAV is limited to an area that has a direct LOS to the control station and is within range of the communication equipment used. Fig. 5 shows this area for an arbitrary position in the environment introduced above.

Fig. 5 Given a position of the control station operator (circle in the middle of the bottom half) the single-hop operational area is limited by range and LOS for the given position, represented by the lightly shaded area. Not all of the mission target's faces (curved building in the middle of the top half, compare Fig. 3) are visible.

In order to complete the reconnaissance task given through the urban first-responder scenario, the operator at some point has to position the primary UAV on the far side of the target building in order to gather detailed data. Under the assumptions of this work, a conventional tactical UAS would be incapable of achieving this as the communication between UAV and operator would be interrupted when the primary UAV leaves the direct LOS area of the GCS operator (Fig. 5).

3.1 Dual-Hop Scenario

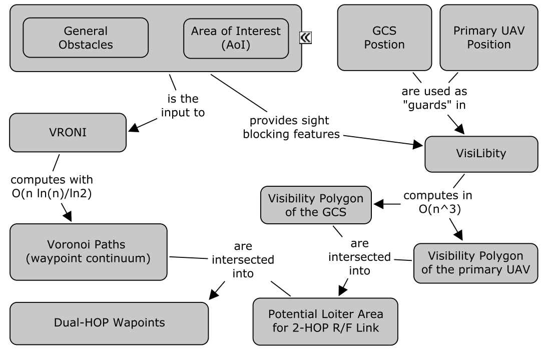
The first proposed method to obtain cues is based on intersecting visibility polygons. Using Obermeyer's VisiLibity ([11, 12]), the visibility polygons for the current position of the GCS and the intended position of the primary UAV are computed based upon the environment data also used for the computation of the Voronoi paths.

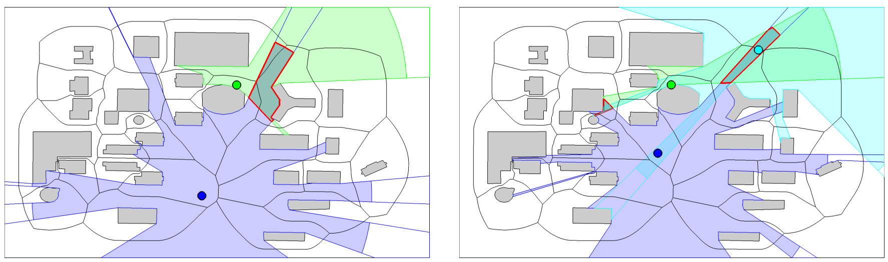
Intersecting these two polygons and the Voronoi paths produces possible way-points which are reachable via translations on the Voronoi paths and also fulfill the requirement to have a direct LOS to the primary UAV as well as the GCS. Fig. 7(a) shows the result of the process outlined in Fig. 6. If applicable, this method only requires one supportive UAV in addition to the tele-operate primary UAV.

3.2 Perimeter Scenario

There are cases in which the dual-hop process proposed in Sec. 3.1 either completely fails, i.e. there is no intersection of the visibility polygons and the Voronoi paths, or

Fig. 6 The process to finding visibility cues for a dual-hop scenario. The Voronoi diagram and the visibility related computations are independent up to the last step of the process. *Guards* is general term for the seeing entities in VisiLibity, here they represent the supportive UAVs.





(a) Outcome of the process: waypoints on the Voronoi paths inside the intersection of the GCS visibility polygons (shaded blue) and the primary UAV's visibility polygon (shaded green) could be used as guidance cues.

(b) A feasible solution that provides very little robustness to motion of the GCS. Indicators are the slender shape of the intersection and the distance of the GCS to the edges of a secondary UAV's visibility polygon (shaded cyan).

Fig. 7 Graphical representation of the dual-hop process results. The outcome of the process is not guaranteed to be usable. Even if the intersection of the visibility polygons and the Voronoi paths are non-empty, the solution might not be robust to movements of either the GCS or the primary UAV.

the resulting cues are not very robust,[5] e.g. as shown in Fig. 7(b). Adapting the dual-hop process (Sec. 3.1) for a larger hop count (i.e. several supportive UAVs) as a main approach to counteract these disadvantages results in a computational load that might not be justifiable as the underlying method seems best suited for single-relay scenarios.

Using the AoI Voronoi perimeter, i.e. the edges of the Voronoi graph that form the segment containing the AoI (see Sec. 2), allows for the computation of a solution that is essentially independent of the positions of the GCS or the primary UAV and provides conceptually a different approach, presumably better suited for multiple relays.

In the process outlined in Fig. 8, the problem of finding cues is translated into the well known "Art Gallery Problem": find the minimum number of guards necessary to observe all walls of an art gallery.

---

5 For computation of visibility robustness see, for example, [13].

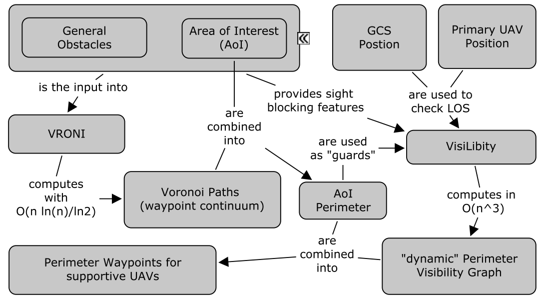


Fig. 8 The process to finding visibility cues for a perimeter scenario. The edges of the Voronoi diagram that also form the edges of the cell containing the mission target form the AoI perimeter.

In the adapted problem, the faces of the target building (the AoI) have to be completely observed and the supportice UAVs (the "guards") can only be located on the Voronoi paths forming the edges of the cell containing the AoI. Additionally, the supportive UAVs, the GCS, and the primary UAV have to be connected in the visibility graph.

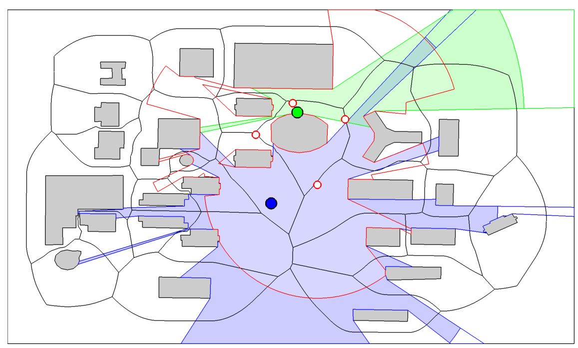


Fig. 9 Graphical representation of the perimeter process results: four guards (red dots) are positioned on the AoI Voronoi perimeter. The guards are connected in the visibility graph and their combined visibility polygon is shown (red outline). As long as the GCS (blue dot) and the primary UAV (green dot) do not leave this polygon, connectedness is ensured.

Fig. 9 shows a result of this process. As long as the GCS and the primary UAV do not leave the combined visibility area of the guards, it is ensured that there always exists a multi-hop connection between the GCS and the primary UAV, using the (stationary) guards as relays.

3.3 Dynamic Visibility

The perimeter process provides cues for a static placement of secondary UAVs for the duration of a mission. However, the perimeter process might also lead to unusable cues, whether due to a limited number of secondary UAVs or due to other constraints, e.g. a resulting congregation of secondary UAVs that exceeds a certain space-density and is hence deemed unsafe. As a next level, a process utilizing the fact that initially only two nodes in the scenario are actually moving could be used to reduce the computational load during a mission, potentially also making use of the *a priori* availability of a map. Fig. 10 outlines this process.

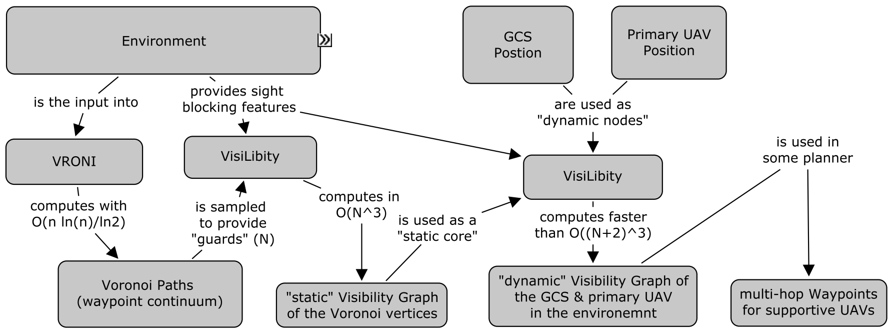


Fig. 10 The process to finding visibility cues for a dynamic scenario. Instead of recomputing the complete visibility graph of the complete environment, the visibility of select points on the Voronoi paths are precomputed (left half of the figure) and later expanded with the visibility information of the two moving nodes, the GCS and the primary UAV. This split in a static/pre-computed part and a dynamic/online-computed part fosters onboard realizability.

As supportive UAVs are limited to positions on the Voronoi paths, the visibility of these potential positions can be precomputed to save computational time during the mission. As the Voronoi paths essentially are a continuum of possible waypoints, a smart sampling has to be developed to reduce the computational effort while keeping a higher resolution where necessary. The edges of the Voronoi path could, for example, be sampled at an increased distance between sample points as the overall distance of the edge to the AoI increases. This would lead to the highest waypoint line-density on the AoI perimeter edges and to lower densities towards the outer areas of the environment.

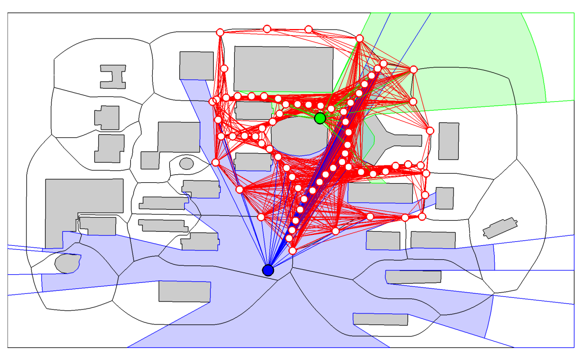


Fig. 11 Using the differentiation of static versus dynamic nodes when computing the visibility speeds up the computation. However, the computational load is still high and the results rather complex. Shown are some nodes of the static core and their respective visibility (red), as well as the visibility of the GCS (blue) and the primary UAV (green).

The result of this process is not inherently geometrical and hence rather hard to visualize. Fig. 11 shows some of the possible paths given by the adjacency matrix of the undirected visibility graph. The process takes the precomputed visibility *static* adjacency matrix $S \in \mathbb{R}^{N\times N}$ of the sampled Voronoi paths and extends it with the *dynamic* adjacency matrix $D \in \mathbb{R}^{N\times 2}$ of nodes representing the GCS and the primary UAV with respect to the the static part. Hence, instead of recomputing the complete visibility graph (with a complexity of $O((N+2)^3)$), the process computes the visibility polygons[6] of the two dynamic nodes (GCS and primary UAV) and checks which of the N sampled points of the Voronoi paths are inside of them. This gives the visibility of the dynamic and the precomputed nodes and the complete adjacency matrix $A \in \mathbb{R}^{(N+2)\times(N+2)}$ can be constructed as $A = \left[\begin{array}{c|c} S & D \\ \hline D^T & \begin{matrix} a_{\mathrm{pUAV}} & 0 \\ 0 & a_{\mathrm{GCS}} \end{matrix} \end{array}\right]$, where $S \in \mathbb{R}^{N\times N}$ is symmetric and $D \in \mathbb{R}^{N\times 2}$.

Any preferred graph algorithm can be used to find paths from the GCS to the primary UAV in the expanded visibility matrix. In Fig. 11 a subset of the complete environment is shown. The potential positions of supportive UAVs are indicated by white circles. The corresponding visibility graph is plotted in red. This would be (part of) the static core. Dynamically computed would be the visibility of the controls station (blue) and the primary UAV (green). As the visibility polygon for both has already been computed in an earlier step, the computation of the actual visibility is reduced to a checking which positions of the static core would be inside this polygon.

4 Conclusions and Remarks

The proposed methods to obtain guidance cues for communication aware UAVs in a cluttered environment are aimed at solving the problem of where to send supportive UAVs to establish a multi-hop communication network between a GCS and a primary UAV. Feasibility of the methods has been tested in simulation for non-moving vehicles, a deployment simulation or actual flight test have not yet been conducted.

The dual-hop process presented in Sec. 3.1 is computationally easy and a very good candidate for onboard implementation. The perimeter process presented in Sec. 3.2 is computationally much more complex, however, as the results are valid for the complete mission (assuming the AoI stays identical) it could be performed *a priori*. The dynamic approach outlined in Sec. 3.3 presents a method to deal with a worst case scenario by dividing potential nodes in two sets, pre-computable static nodes and moving dynamic ones. As this process would only be reached if the other presented processes fail, previously computed data can be reused to minimize the computational impact.

Although the dynamic process seems to implementable onboard (from a computational perspective), this process also poses the most challenges in transforming the cues into actually selected target locations for secondary UAVs. As the result of

[6] In a staged approach of trying the proposed processes in the presented order, these can be reused from the dual-hop process.

the process is just an adjacency matrix (where shortest path graph algorithms can give cues for multi-hop waypoints), additional metrics have to be found or defined in order to rank the set of shortest (hop) paths between primary UAV and the GCS, assuming that fewer supportive UAVs are preferable.

The proposed methods however do not yet deal with some of the immediately imminent challenges: as the primary UAV moves unpredictably for the automation (as it is tele-operated), the automation has to anticipate its motion and pre-plan for all possibilities. This might lead to conflicting requirements for the positioning of the secondary UAV(s) when the guidance has to sort out where to send them. Future work will have to look into the challenges resulting therefrom.

Though experimentation with realistic urban scenarios seems to indicate that the dual-hop process most often leads to usable results (where the use of some notion of visibility robustness can rank the cues given by the process in order to obtain definite waypoint for the underlying guidance), more simulation - particularly of the deployment - is necessary to provide usable and implementable heuristics and algorithms.

References

1. E. Salas, *Human Factors of Remotely Operated Vehicles*, *Advances in Human Performance and Cognitive Engineering Research*, vol. 7, 1st edn. (Emerald Group Publishing Limited, 2006). URL `http://books.google.com/books?id=ZRHzDothAqAC`. Editors: Nancy J. Cooke, Heather L. Pringle, Harry K. Pedersen, Olena Connor
2. H.C. Christmann, E.N. Johnson, in *AIAA Modeling and Simulation Technologies Conference and Exhibit* (2010). AIAA-2010-8361
3. D.T. Wooden, Graph-based path planning for mobile robots. Ph.D. thesis, Georgia Institute of Technology (2006). URL `http://hdl.handle.net/1853/14055`
4. S. König, M. Likhachev, IEEE Transactions on Robotics **21**(3), 354 (2005). DOI 10.1109/TRO.2004.838026
5. M. Held, Computational Geometry **18**(2), 95 (2001). DOI 10.1016/S0925-7721(01)00003-7. URL `http://www.sciencedirect.com/science/article/B6TYS-42VM7PV-3/2/6d5a871c35c57105b135f211f29f4f89`
6. M.D. Sobers, Efficient ranging-sensor navigation methods for indoor aircraft. Ph.D. thesis, Georgia Institute of Technology (2010). URL `http://hdl.handle.net/1853/34824`
7. A. Jenkins, D. Henkel, T. Brown, in *Proceedings on Infotech@Aerospace 2007 Conference and Exhibit* (AIAA, Rohnert Park, California, United States, 2007). URL `http://pecolab.colorado.edu/papers/jenkins-AIAAsdc.pdf`. AIAA-2007-2747
8. E. Frew, T. Brown, Proceedings of the IEEE **96**(12), 2008 (2008). DOI 10.1109/JPROC.2008.2006127
9. S. Hauert, S. Leven, J.C. Zufferey, D. Floreano, in *Robotics and Automation (ICRA), 2010 IEEE International Conference on* (2010), pp. 15 –20. DOI 10.1109/ROBOT.2010.5509421
10. H.C. Christmann, E.N. Johnson, in *AIAA Modeling and Simulation Technologies Conference and Exhibit* (Honolulu, HI, USA, 2008). AIAA-2008-7042
11. K.J. Obermeyer, Contributors. The VisiLibity library (2008). URL `http://www.VisiLibity.org`. R-1
12. K.J. Obermeyer, Visibility problems for sensor networks and unmanned air vehicles. Ph.D. thesis, Mechanical Engineering Department, University of California at Santa Barbara (2010). URL `http://ccdc.mee.ucsb.edu`

13. B. Ben-Moshe, O. Hall-Holt, M.J. Katz, J.S.B. Mitchell, in *SCG '04: Proceedings of the twentieth annual symposium on Computational geometry* (ACM, New York, NY, USA, 2004), pp. 27–35. DOI 10.1145/997817.997825